ELSEVIER Linear Algebra and its Applications 296 (1999) 113–141

LINEAR ALGEBRA AND ITS APPLICATIONS

www.elsevier.com/locate/laa

Multiparameter descent methods

C. Brezinski *

Laboratoire d'Analyse Numérique et d'Optimisation, Université des Sciences et Technologies de Lille, 59655-Villeneuve d'Ascq Cedex, France

Received 31 January 1999; accepted 14 May 1999

Submitted by R.A. Brualdi

Abstract

A descent method for solving a system of linear equations $Ax = b$ consists of the iterations $x_{k+1} = x_k + \lambda_k z_k$, where z_k is a vector and λ_k a parameter chosen to minimize some functional. In this paper, we will consider multiparameter generalizations of such descent methods, namely iterations of the form $x_{k+1} = x_k + Z_k \Lambda_k$, where Z_k is a matrix and Λ_k a vector chosen to minimize some functional. Multiparameter generalizations of the conjugate direction method and the Lanczos method will be obtained and their algebraic properties discussed. Multiparameter conjugate and biconjugate gradient algorithms and other Lanczos-type algorithms for implementing this Lanczos method will also be given. © 1999 Elsevier Science Inc. All rights reserved.

Keywords: Linear systems; Lanczos method; Conjugate gradient; Descent method

0. Introduction

Many methods for solving a system of linear equations $Ax = b$, or a system of nonlinear equations $f(x) = 0$, or for minimizing a convex functional J consist of iterations of the form

$$x_{k+1} = x_k + \lambda_k z_k, \quad r_{k+1} = r_k - \lambda_k A z_k, \tag{1}$$

where z_k is a vector called the *direction of descent*, λ_k a parameter, the *stepsize*, chosen to minimize some quantity, and $r_k = b - Ax_k$ the residual.

* Tel.: +33-3-2043-4296; fax: +33-3-2043-6869; e-mail: Claude.Brezinski@univ-lille1.fr

PII: S 0 0 2 4 - 3 7 9 5 (9 9) 0 0 1 1 2 - 3

Among these methods are the classical relaxation methods, projection methods and the steepest descent method [4], the method of Barzilai and Borwein [1], the method of Richardson [2] and some others as well; see [3] for details about these methods.

In this paper, instead of considering descent methods involving only one parameter λ_k, we will consider iterations of the form

$$x_{k+1} = x_k + Z_k \Lambda_k, \quad r_{k+1} = r_k - AZ_k\Lambda_k, \tag{2}$$

where Z_k is a full rank $n \times n_k$ matrix, called the *descent matrix*, n is the dimension of the system, $n_k \leqslant n$ and Λ_k a vector of $\mathbb{R}^{n_k}$ called the *stepsize vector*. The vector Λ_k will be chosen in order to minimize some quantity and, so, the iterative method given by (2) appears as a *multiparameter descent method*.

In this paper, we will discuss iterative methods of the form (2) and consider three different minimization criteria for computing the vector Λ_k. They correspond to three different types of methods, namely:

1. Multiparameter Richardson method

$$\|r_{k+1} = b - Ax_{k+1}\| = \min.$$

2. Multiparameter projection method

$$J(x_{k+1}) = \tfrac{1}{2}(x_{k+1}, Ax_{k+1}) - (x_{k+1}, b) = \min.$$

3. Multiparameter Barzilai–Borwein method

$$\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\| = \min.$$

where all norms are the Euclidean one. Obviously, the solution of these three minimization problems is related to the Moore–Penrose pseudo-inverse of a rectangular matrix (see, for example, [10]).

The first and the second minimization criteria only apply to systems of linear equations and, moreover, in the second case, the matrix A has to be symmetric and positive definite. The third criterion is valid either for systems of linear or nonlinear equations. Other possible criteria will also be given.

Several choices of the matrices Z_k are of interest.

First, the optimal choice for the matrix Z_k will be discussed and a partitioning strategy will be proposed. This strategy corresponds, in fact, to iterations of the form (1), but with a different parameter λ_k for each component (or block of components) of the vector z_k. Indeed, in some cases, when solving systems of linear or nonlinear equations by an iterative method, the components (or blocks of components) of the iterates do not at all exhibit the same behavior. This is the case, for example, in decomposition methods, multigrid methods, wavelets, multiresolution, inertial manifolds, incremental unknowns, the nonlinear Galerkin method, synchronous and asynchronous iterations, or when the components of the solution represent different physical quantities, … So, in such situations, a gain will be observed by splitting the iterates into blocks of components having a quite similar behavior and treating them differently. The partitioning strategy addresses this problem. It was introduced in [7] where some vector sequence transformations were proposed and their efficiency illustrated by several numerical examples.

Then, a multiparameter conjugate direction method will be proposed. In this method, choosing the matrices Z_k conjugate with respect to A leads to a multiparameter Lanczos method. For the implementation of this Lanczos method, a multiparameter generalization of the conjugate gradient algorithms will be given. Then, the nonsymmetric case will be considered and multiparameter biconjugate gradient algorithms, and some other algorithms as well, will be derived and their algebraic properties will be given.

Before studying these cases, let us first explain how relaxation methods fit into iterations of the form (1) and (2). We consider the nonsingular linear system $Ax = b$. Let C_k be an approximation of A^{-1}, that is a preconditioner. We set $T_k = I - C_kA$ and $d_k = C_kb$. The left preconditioned system $C_kAx = d_k$ can be written as $x = T_kx + d_k$ and we consider the relaxation method given by

$$\begin{aligned} x_{k+1} &= T_kx_k + d_k \\ &= (I - C_kA)x_k + C_kb \\ &= x_k + C_kr_k \end{aligned}$$

with $r_k = b - Ax_k$. So, these iterations correspond to the iterations (1) if λ_kz_k is such that $\lambda_kz_k = C_kr_k$. Similarly, for the multiparameter iterations (2), $Z_k\Lambda_k$ has to satisfy $Z_k\Lambda_k = C_kr_k$.

Let us mention that the idea leading to multiparameter descent methods can also be used for transforming a sequence (x_k) obtained by any iterative method into a new sequence (y_k) having, under some assumptions, a better convergence behavior than (x_k). We set

$$y_k = x_k + Z_k\Lambda_k, \quad \rho_k = b - Ay_k = r_k - AZ_k\Lambda_k$$

and, as above, we choose Λ_k either such that $\|\rho_k\|$ is minimum, or such that $J(y_k)$ is minimum, or such that $\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\|$ is minimum. Some sequence transformations of this type were already studied in [8, 5] where they were shown to be quite effective. They are called *multiparameter acceleration methods*.

Remark 1. The case of the left preconditioned system $C^{-1}Ax = C^{-1}b$ can be treated by replacing A by $C^{-1}A$ and r_k by $C^{-1}r_k$, where r_k is still defined by $r_k = b - Ax_k$. The case of the right preconditioned system $AC^{-1}x' = b$, with $x = C^{-1}x'$, can be treated by replacing A by AC^{-1} and x_k by $C^{-1}x_k$, the residuals r_k remaining unchanged. Both strategies can be combined together in the case of $C^{-1}AC'^{-1}x' = C^{-1}b$ with $x = C'^{-1}x'$.

Before going farther, we need to introduce a generalization of the notion of Schur complement. Let $v \in \mathbb{R}^n$, $d \in \mathbb{R}^m$, $V = [v_1, \ldots, v_m]$ with $v_i \in \mathbb{R}^n$ and D a $m \times m$ nonsingular matrix. We have the following identity which generalizes Schur's

$$v - VD^{-1}d = \frac{\begin{vmatrix} v & V \\ d & D \end{vmatrix}}{|D|}, \tag{3}$$

where the determinant in the numerator denotes the vector of $\mathbb{R}^n$ obtained by expanding it with respect to its first (vector) row, treated as a row of scalars, by the classical rule for expanding a determinant. For example, if $m = 1$, we have

$$\begin{vmatrix} v & v_1 \\ d & d_1 \end{vmatrix} = d_1 v - d v_1.$$

On this subject, see [5].

1. The methods

We will now examine the multiparameter descent methods obtained by the three preceding minimization criteria. In the first two cases, we consider a system of linear equations $Ax = b$ where A is a square nonsingular matrix. The third case can be applied either to a system of linear or nonlinear equations. Finally, we will also consider some other possible choices which apply to the solution of a nonsingular linear system.

1.1. The multiparameter Richardson method

The vector Λ_k will be chosen in order to minimize

$$\|r_{k+1}\|, \quad \text{where} \quad r_{k+1} = b - Ax_{k+1}.$$

We have $r_{k+1} = r_k - AZ_k\Lambda_k$. Thus, we must choose Λ_k so that $r_k - AZ_k\Lambda_k = 0$ in the least squares sense, that is

$$\Lambda_k = \left[(AZ_k)^{\mathrm{T}} AZ_k\right]^{-1} (AZ_k)^{\mathrm{T}} r_k.$$

Such a method was introduced in [7] as a multiparameter variant of the Richardson iterative method discussed in [2] (see also [6,7] where quite similar ideas are developed).

It holds $Z_k^{\mathrm{T}} A r_{k+1} = 0$. Moreover, setting $P_k = U_k(U_k^{\mathrm{T}} U_k)^{-1} U_k^{\mathrm{T}}$ with $U_k = AZ_k$, we see that $P_k = P_k^2$ and $P_k = P_k^{\mathrm{T}}$ which shows that P_k is an orthogonal projection. It is the orthogonal projection on the subspace spanned by the columns of the matrix U_k.

With this optimal choice of Λ_k, we have

$$\|r_{k+1}\|^2 = \|r_k\|^2 - (r_k, P_k r_k) = \|r_k\|^2 - \|P_k r_k\|^2 \leqslant \|r_k\|^2.$$

Using formula (3), we obtain

$$x_{k+1} = \frac{\begin{vmatrix} x_k & -Z_k \\ (AZ_k)^{\mathrm{T}} r_k & (AZ_k)^{\mathrm{T}} AZ_k \end{vmatrix}}{\left|(AZ_k)^{\mathrm{T}} AZ_k\right|}, \qquad r_{k+1} = \frac{\begin{vmatrix} r_k & AZ_k \\ (AZ_k)^{\mathrm{T}} r_k & (AZ_k)^{\mathrm{T}} AZ_k \end{vmatrix}}{\left|(AZ_k)^{\mathrm{T}} AZ_k\right|}.$$

At the step k, let us assume that, instead of computing r_{k+1} from the matrix Z_k, we compute r'_{k+1} from the bordered matrix $Z'_k = [Z_k, y]$ where y is an additional vector. Thus, thanks to the minimization property of the stepsize vector, we have

$$\begin{aligned} \|r'_{k+1}\| &= \min_{\Lambda} \|r_k - AZ'_k \Lambda\| \leqslant \|r_k - A[Z_k, y](\Lambda_k^{\mathrm{T}}, 0)^{\mathrm{T}}\| \\ &= \|r_k - AZ_k \Lambda_k\| = \|r_{k+1}\|. \end{aligned}$$

After some calculations, it can be proved that the gain obtained by bordering the matrix Z_k by a new vector y is given by

$$\|r_{k+1}\|^2 - \|r'_{k+1}\|^2 = \frac{(r_k, (I - P_k)Ay)^2}{(Ay, (I - P_k)Ay)}.$$

Since $I - P_k$ is symmetric positive semi-definite, the denominator in this expression is positive and we obtain $\|r'_{k+1}\| \leqslant \|r_{k+1}\|$.

1.2. The multiparameter projection method

For this case, we assume that the matrix A is symmetric positive definite. It is well known that $x = A^{-1}b$ is the only vector which minimizes

$$J(u) = \tfrac{1}{2}(u, Au) - (u, b).$$

Thus, the vector Λ_k will be chosen in order to minimize $J(x_{k+1})$. We have

$$J(x_{k+1}) = J(x_k) - (Z_k \Lambda_k, r_k) + \tfrac{1}{2}(Z_k \Lambda_k, AZ_k \Lambda_k).$$

The functional J is minimum when its partial derivatives with respect to the components λ_i of Λ_k are zero. We have

$$\begin{aligned} &\frac{\partial}{\partial \lambda_i}(Z_k \Lambda_k, r_k) = (z_i, r_k), \\ &\frac{\partial}{\partial \lambda_i}(Z_k \Lambda_k, AZ_k \Lambda_k) = 2\sum_{j=1}^{n_k} \lambda_j (z_i, Az_j), \end{aligned}$$

where z_i denotes the ith column of the matrix Z_k. Thus, we must have

$$\frac{\partial}{\partial \lambda_i} J(x_{k+1}) = \sum_{j=1}^{n_k} \lambda_j (z_i, Az_j) - (z_i, r_k) = 0, \quad i = 1, \ldots, n_k$$

that is, in other terms,

$$Z_k^{\mathrm{T}} A Z_k \Lambda_k = Z_k^{\mathrm{T}} r_k,$$

which leads to

$$\Lambda_k = (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} r_k. \tag{4}$$

With this optimal choice of Λ_k, we have

$$J(x_{k+1}) = J(x_k) - \tfrac{1}{2} (Z_k^{\mathrm{T}} r_k, (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} r_k) \leqslant J(x_k)$$

since the matrix $Z_k^{\mathrm{T}} A Z_k$ is symmetric positive definite. It also holds $Z_k^{\mathrm{T}} r_{k+1} = 0$. Setting $e_k = x - x_k$, we have $e_{k+1} = e_k - Z_k \Lambda_k$. Thus, with the notation $\|e_k\|_A^2 = (e_k, Ae_k)$, we have

$$\|e_{k+1}\|_A^2 = \|e_k\|_A^2 - (Z_k \Lambda_k, Ae_k) - (e_k, AZ_k \Lambda_k) + (Z_k \Lambda_k, AZ_k \Lambda_k).$$

Since $A = A^{\mathrm{T}}$ and $r_k = Ae_k$, the preceding expression becomes

$$\|e_{k+1}\|_A^2 = \|e_k\|_A^2 - 2(Z_k \Lambda_k, r_k) + (Z_k \Lambda_k, AZ_k \Lambda_k)$$

and it is easy to see that our choice for Λ_k also minimizes $\|e_{k+1}\|_A$ and that, for this value of Λ_k, $\|e_{k+1}\|_A^2 = \|e_k\|_A^2 - ((Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} r_k, Z_k^{\mathrm{T}} r_k)$. So, $\|e_{k+1}\|_A \leqslant \|e_k\|_A$. Such an approach to multiparameter projection methods was already considered in [20] and [19, pp. 98–103] (see also [3, p. 276]).

Since A is symmetric positive definite, there exists a matrix B such that $A = B^{\mathrm{T}} B$. The vector Λ_k is also the solution of the system $Be_{k+1} = 0$ in the least squares sense. Indeed, $Be_{k+1} = Be_k - BZ_k \Lambda_k$ and, so, the least squares solution is given by

$$\begin{aligned} \Lambda_k &= [(BZ_k)^{\mathrm{T}} BZ_k]^{-1} (BZ_k)^{\mathrm{T}} Be_k \\ &= (Z_k^{\mathrm{T}} B^{\mathrm{T}} BZ_k)^{-1} Z_k^{\mathrm{T}} B^{\mathrm{T}} Be_k \\ &= (Z_k^{\mathrm{T}} AZ_k)^{-1} Z_k^{\mathrm{T}} Ae_k = (Z_k^{\mathrm{T}} AZ_k)^{-1} Z_k^{\mathrm{T}} r_k. \end{aligned}$$

If we set $P_k = AZ_k (Z_k^{\mathrm{T}} AZ_k)^{-1} Z_k^{\mathrm{T}}$, then $P_k^2 = P_k$ but $P_k^{\mathrm{T}} \neq P_k$. Thus P_k represents the oblique projection on the subspace spanned by the columns of AZ_k orthogonally to the subspace spanned by the columns of Z_k. We set

$$\begin{aligned} Q_k &= BZ_k (Z_k^{\mathrm{T}} AZ_k)^{-1} Z_k^{\mathrm{T}} B^{\mathrm{T}} \\ &= BZ_k (Z_k^{\mathrm{T}} B^{\mathrm{T}} BZ_k)^{-1} Z_k^{\mathrm{T}} B^{\mathrm{T}} \\ &= V_k (V_k^{\mathrm{T}} V_k)^{-1} V_k^{\mathrm{T}} \end{aligned}$$

with $V_k = BZ_k$. We have $Q_k^2 = Q_k$ and $Q_k^{\mathrm{T}} = Q_k$. Thus, Q_k is the orthogonal projection on the subspace spanned by the columns of the matrix BZ_k and we have $P_k = B^{\mathrm{T}} Q_k B^{-\mathrm{T}}$.

Using formula (3), we obtain

$$x_{k+1} = \frac{\begin{vmatrix} x_k & -Z_k \\ Z_k^{\mathrm{T}} r_k & Z_k^{\mathrm{T}} A Z_k \end{vmatrix}}{|Z_k^{\mathrm{T}} A Z_k|}, \qquad r_{k+1} = \frac{\begin{vmatrix} r_k & AZ_k \\ Z_k^{\mathrm{T}} r_k & Z_k^{\mathrm{T}} A Z_k \end{vmatrix}}{|Z_k^{\mathrm{T}} A Z_k|}.$$

Since the symmetry of the matrix A is the only property which is required in the preceding expressions involving scalar products, the multiparameter projection method can be applied to systems with a symmetric but not necessarily positive definite matrix. However, it must be noticed that, in this case, J is no longer convex and, so, the minimization and monotonicity properties are not always satisfied. If A is not symmetric, there exists several strategies for replacing the system $Ax = b$ by a symmetric one having the same solution. These strategies are described in [3, p. 97] or in [4]. They will be discussed in Section 6.

At the step k, let us assume that, instead of computing $J(x_{k+1})$ from the matrix Z_k, we compute $J(x'_{k+1})$ from the bordered matrix $Z'_k = [Z_k, y]$ where y is again an additional vector. Thus, thanks to the minimization property of the stepsize vector, we have

$$\begin{aligned} J(x'_{k+1}) &= \min_{\Lambda} J(x_k + Z'_k \Lambda) \leqslant J(x_k + [Z_k, y](\Lambda_k^{\mathrm{T}}, 0)^{\mathrm{T}}) \\ &= J(x_k + Z_k \Lambda_k) = J(x_{k+1}). \end{aligned}$$

After some calculations, it can be proved that the gain obtained by bordering the matrix Z_k by a new vector y is given by

$$J(x_{k+1}) - J(x'_{k+1}) = \frac{[(S_k r_k, Ay) - (r_k, y)]^2}{2(y, (I - AS_k)Ay)},$$

where $S_k = Z_k (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}}$. Since $A = B^{\mathrm{T}} B$, the scalar product in the denominator is equal to $(By, (I - BZ_k (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} B^{\mathrm{T}})By)$ which is nonnegative since the matrix appearing there is symmetric positive semi-definite. Thus $J(x'_{k+1}) \leqslant J(x_{k+1})$.

1.3. The multiparameter Barzilai–Borwein method

For this method, it is mandatory to assume that, for all k, n_k is constant. The vector Λ_k is chosen to minimize

$$\|\Delta x_{k-1} + \Delta Z_{k-1} \Lambda_k\|.$$

Such a Λ_k must satisfy $\Delta x_{k-1} + \Delta Z_{k-1} \Lambda_k = 0$ in the least squares sense, which gives

$$\Lambda_k = -\left[(\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\right]^{-1}(\Delta Z_{k-1})^{\mathrm{T}}\Delta x_{k-1}.$$

With this optimal choice of Λ_k, we have

$$\begin{aligned}&\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\|^2\\ &\quad = \|\Delta x_{k-1}\|^2 - \left(\Delta x_{k-1}, \Delta Z_{k-1}\left[(\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\right]^{-1}(\Delta Z_{k-1})^{\mathrm{T}}\Delta x_{k-1}\right)\end{aligned}$$

and so $\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\| \leqslant \|\Delta x_{k-1}\|$ since $(\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}$ is symmetric positive definite.

If we set $P_k = U_k(U_k^{\mathrm{T}}U_k)^{-1}U_k^{\mathrm{T}}$ with $U_k = \Delta Z_{k-1}$, then $P_k^2 = P_k$ and $P_k^{\mathrm{T}} = P_k$ which shows that P_k is the orthogonal projection on the subspace spanned by the columns of the matrix ΔZ_{k-1}.

This method is a multiparameter generalization of the method proposed by Barzilai and Borwein [1], whose convergence was studied by Raydan [24] and a preconditioned version was given by Molina and Raydan [22].

Using formula (3), we obtain

$$x_{k+1} = \frac{\begin{vmatrix} x_k & Z_k \\ (\Delta Z_{k-1})^{\mathrm{T}}\Delta x_{k-1} & (\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\end{vmatrix}}{\left|(\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\right|},$$

$$r_{k+1} = \frac{\begin{vmatrix} r_k & -AZ_k \\ (\Delta Z_{k-1})^{\mathrm{T}}\Delta x_{k-1} & (\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\end{vmatrix}}{\left|(\Delta Z_{k-1})^{\mathrm{T}}\Delta Z_{k-1}\right|}.$$

1.4. Other methods

For solving a system of nonsingular linear equations, there are other possible choices for the parameter λ_k in the one-parameter case. Let u_k be an arbitrary vector. One can choose λ_k such that $(u_k, r_{k+1}) = 0$ which leads to $\lambda_k = (u_k, r_k)/(u_k, Az_k)$. Taking $u_k = r_k$ seems to be new. The corresponding multiparameter method consists of taking Λ_k such that $U_k^{\mathrm{T}}r_{k+1} = 0$ where U_k is an arbitrary matrix. We get

$$U_k^{\mathrm{T}}r_{k+1} = U_k^{\mathrm{T}}r_k - U_k^{\mathrm{T}}AZ_k\Lambda_k = 0$$

which gives

$$\Lambda_k = (U_k^{\mathrm{T}}AZ_k)^{-1}U_k^{\mathrm{T}}r_k$$

and we obtain

$$x_{k+1} = \frac{\begin{vmatrix} x_k & -Z_k \\ U_k^{\mathrm{T}} r_k & U_k^{\mathrm{T}} A Z_k \end{vmatrix}}{|U_k^{\mathrm{T}} A Z_k|}, \qquad r_{k+1} = \frac{\begin{vmatrix} r_k & A Z_k \\ U_k^{\mathrm{T}} r_k & U_k^{\mathrm{T}} A Z_k \end{vmatrix}}{|U_k^{\mathrm{T}} A Z_k|}.$$

For the choice $U_k = Z_k$, the multiparameter projection method is recovered.

In the one-parameter case, another possible procedure consists of setting $x_{k+1} = x_k + \lambda_k z_k$ and choosing λ_k so that $(u_{k-1}, \Delta x_{k-1} + \lambda_k \Delta z_{k-1}) = 0$, that is $\lambda_k = -(u_{k-1}, \Delta x_{k-1})/(u_{k-1}, \Delta z_{k-1})$. For the multiparameter case of this procedure (assuming, again, that $\forall k, n_k$ is constant), we set $x_{k+1} = x_k + Z_k \Lambda_k$ with Λ_k so that $U_{k-1}^{\mathrm{T}}(\Delta x_{k-1} + \Delta Z_{k-1} \Lambda_k) = 0$, that is

$$\Lambda_k = -(U_{k-1}^{\mathrm{T}} \Delta Z_{k-1})^{-1} U_{k-1}^{\mathrm{T}} \Delta x_{k-1}.$$

Thus, we obtain the iterations

$$x_{k+1} = x_k - Z_k (U_{k-1}^{\mathrm{T}} \Delta Z_{k-1})^{-1} U_{k-1}^{\mathrm{T}} \Delta x_{k-1}$$

and it follows:

$$x_{k+1} = \frac{\begin{vmatrix} x_k & Z_k \\ U_{k-1}^{\mathrm{T}} \Delta x_{k-1} & U_{k-1}^{\mathrm{T}} \Delta Z_{k-1} \end{vmatrix}}{\left|U_{k-1}^{\mathrm{T}} \Delta Z_{k-1}\right|}, \qquad r_{k+1} = \frac{\begin{vmatrix} r_k & -A Z_k \\ U_{k-1}^{\mathrm{T}} \Delta x_{k-1} & U_{k-1}^{\mathrm{T}} \Delta Z_{k-1} \end{vmatrix}}{\left|U_{k-1}^{\mathrm{T}} \Delta Z_{k-1}\right|}.$$

This method is closely related to the VTT introduced in [8] and to the methods based on the Θ-strategy discussed in [5]. If U_{k-1} is such that $U_{k-1}^{\mathrm{T}} Z_k = 0$, we obtain an iterative method in the style of the BVTT given in [8]. For the choice $U_{k-1} = \Delta Z_{k-1}$, the method of Barzilai–Borwein is recovered.

It is also possible to derive iterative procedures in the style of the E, $E\Theta$ and ΘE strategies introduced in [5] for the construction of vector sequence transformations.

2. Choice of the descent matrix

Let us assume that, at the iteration k, Z_k can be chosen so that $\exists \alpha_k \in \mathbb{R}^{n_k}$ satisfying $r_k = A Z_k \alpha_k$ that is, in other terms, $x = x_k + Z_k \alpha_k$ or

$$Z_k \alpha_k = A^{-1} r_k.$$

From the determinantal identities given in the preceding section, we see that, with such a Z_k, all the multiparameter methods will give $x_{k+1} = x$.

Since the preceding choice is not possible in practice, we will choose Z_k so that $\exists \alpha_k$ satisfying

$$Z_k \alpha_k = C_k r_k, \tag{5}$$

where C_k is an approximation of A^{-1}, that is, in other words, a preconditioner.

For the multiparameter Richardson method, since the choice of Λ_k minimizes $\|r_{k+1}\|$, we have, for any vector α_k, $\|r_{k+1}\| \leqslant \|r_k - AZ_k\alpha_k\|$. In particular, if $Z_k\alpha_k = C_kr_k$, then we obtain

$$\|r_{k+1}\| \leqslant \|r_k - AC_kr_k\| \leqslant \|I - AC_k\| \cdot \|r_k\|.$$

Since C_k is an approximation of A^{-1}, $\|I - C_kA\|$ is usually small and this inequality shows the improvement brought. Moreover, if the sequence (C_k) tends to A^{-1}, then we see that the method converges superlinearly.

For the multiparameter projection method, since the choice of Λ_k minimizes $J(x_{k+1})$, we have, for any vector α_k,

$$0 \leqslant J(x_{k+1}) - J(x) \leqslant J(x_k + Z_k\alpha_k) - J(x).$$

In particular, if $Z_k\alpha_k = C_kr_k$, we obtain

$$\begin{aligned} J(x_k + Z_k\alpha_k) - J(x) &= J(x_k + C_kr_k) - J(x) \\ &= \tfrac{1}{2}((I - C_kA)e_k, A(I - C_kA)e_k), \end{aligned}$$

which shows the gain brought and that the convergence is fast if (C_k) tends to A^{-1}.

For the multiparameter Barzilai–Borwein method, because of the minimizing property of Λ_k, we have, for any constant vector $\alpha_k = \alpha$ (which will be the case below),

$$\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\| \leqslant \|\Delta x_{k-1} + \Delta Z_{k-1}\alpha\| = \|\Delta x_{k-1} + \Delta(C_{k-1}r_{k-1})\|.$$

But $\Delta x_{k-1} = -\Delta e_{k-1}$, $r_{k-1} = Ae_{k-1}$ and we finally obtain

$$\|\Delta x_{k-1} + \Delta Z_{k-1}\Lambda_k\| \leqslant \|\Delta((I - C_{k-1}A)e_{k-1})\|,$$

which shows the improvement brought and the fast convergence of the method if (C_k) tends to A^{-1}.

Let us now see how to construct a matrix Z_k satisfying the condition (5). We will follow the partitioning strategy introduced in [7]. Let $z_k \in \mathbb{R}^n$ be an arbitrary vector. We partition it into n_k blocks of components $z_k = (z_k^{1^{\mathrm{T}}}, \ldots, z_k^{n_k^{\mathrm{T}}})^{\mathrm{T}}$ with $z_k^i \in \mathbb{R}^{p_i}$ and $p_1 + \cdots + p_{n_k} = n$, and we consider the $n \times n_k$ matrix

$$Z_k = \begin{pmatrix} z_k^1 & & \\ & \ddots & \\ & & z_k^{n_k} \end{pmatrix}. \tag{6}$$

Setting $\Lambda_k = (\lambda_k^1, \ldots, \lambda_k^{n_k})^{\mathrm{T}}$, we see that $Z_k\Lambda_k = (\lambda_k^1 z_k^{1^{\mathrm{T}}}, \ldots, \lambda_k^{n_k} z_k^{n_k^{\mathrm{T}}})^{\mathrm{T}} \in \mathbb{R}^n$. So, this choice corresponds to taking a different parameter λ_k^i for each block of components z_k^i instead of the same parameter for all the components, as it is the case in one-parameter methods. As already mentioned in [7], it is important to notice that $Z_k\Lambda_k = \Gamma_kz_k$, where Γ_k is the $n \times n$ matrix

$$\Gamma_k = \begin{pmatrix} \lambda_k^1 I_1 & & \\ & \ddots & \\ & & \lambda_k^{n_k} I_{n_k} \end{pmatrix}$$

and I_i is the identity matrix of dimension p_i.

Let us partition the vector $C_k r_k$ into n_k blocks $(C_k r_k)^i \in \mathbb{R}^{p_i}$ and take $z_k^i = (C_k r_k)^i$. Let $e = (1, \ldots, 1)^{\mathrm{T}} \in \mathbb{R}^{n_k}$. Then, we have $Z_k e = z_k = C_k r_k$ which shows that this choice for Z_k satisfies (5) with, $\forall k$, $\alpha_k = e$.

Obviously, a more complicated partitioning strategy can be used. It consists of selecting one (and only one) different component of z_k for each row of Z_k, all the other components being zero. For example, if $n = 6$ and $n_k = 3$

$$Z_k = \begin{pmatrix} * & 0 & 0 \\ 0 & * & 0 \\ * & 0 & 0 \\ 0 & 0 & * \\ 0 & * & 0 \\ 0 & * & 0 \end{pmatrix},$$

where a $*$ denotes a component of the vector z_k.

In that case, we still have $Z_k e = z_k$. However, it must be noticed that a matrix of the form (6) can be recovered by permuting the equations.

Taking $z_k = r_k$ and constructing Z_k by partitioning z_k can be considered as a *preconditioned multiparameter steepest descent* method.

3. The multiparameter conjugate direction method

Let us now come back to the case where A is symmetric positive definite and consider another way of choosing the descent matrices Z_k in the multiparameter projection method. In this approach, we will adapt the arguments used in the one-parameter case (see for example, [15, p. 272; 3, pp. 108–112] or [4]).

Let $E_k = \mathrm{span}(Z_0, \ldots, Z_k)$ be the subspace of $\mathbb{R}^n$ spanned by the columns of the $n \times n_i$ matrices Z_i for $i = 0, \ldots, k$. Thus, the dimension of E_k is $m_k = n_0 + \cdots + n_k$. We will assume that $m_k \leqslant n$ and look for the vector $x_{k+1} \in E_k$, of the form $x_{k+1} = x_k + Z_k \alpha$ with $x_0 = 0$, which minimizes $J(x_{k+1})$. Since $x_0 = 0$, such a vector x_{k+1} can be written as $x_{k+1} = W_k w + Z_k \alpha$ with $W_k = [Z_0, \ldots, Z_{k-1}] \in \mathbb{R}^{n \times m_{k-1}}$, $w \in \mathbb{R}^{m_{k-1}}$ and $\alpha \in \mathbb{R}^{n_k}$. So, we have to minimize $J(x_{k+1})$ with respect to w and α. We get

$$\begin{aligned} J(x_{k+1}) &= \tfrac{1}{2}(W_k w + Z_k \alpha, A W_k w + A Z_k \alpha) - (b, W_k w + Z_k \alpha) \\ &= \tfrac{1}{2}(W_k w, A W_k w) - (b, W_k w) + \tfrac{1}{2}(Z_k \alpha, A W_k w) \\ &\quad + \tfrac{1}{2}(W_k w, A Z_k \alpha) + \tfrac{1}{2}(Z_k \alpha, A Z_k \alpha) - (b, Z_k \alpha) \\ &= J(W_k w) + (W_k w, A Z_k \alpha) + \tfrac{1}{2}(Z_k \alpha, A Z_k \alpha) - (b, Z_k \alpha). \end{aligned}$$

The minimization problem is made more complicated by the cross term $(W_k w, AZ_k \alpha)$. Without it, the minimization decouplates into a minimization over the range of W_k (whose solution is already known and is x_k) and a minimization problem with respect to the vector α. Thus, assuming that

$$(W_k w, AZ_k \alpha) = 0 \quad \forall w, \alpha \tag{7}$$

we have

$$\min_{w,\alpha} J(x_{k+1}) = \min_{w} J(W_k w) + \min_{\alpha} \left[\tfrac{1}{2}(Z_k \alpha, AZ_k \alpha) - (b, Z_k \alpha)\right].$$

The solution of this minimization problem is given by

$$w = (\Lambda_0^{\mathrm{T}}, \ldots, \Lambda_{k-1}^{\mathrm{T}})^{\mathrm{T}}, \qquad \alpha = (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} b.$$

But, since $x_k \in E_{k-1}$, we have $x_k = W_k w$ and $Z_k^{\mathrm{T}} A x_k = Z_k^{\mathrm{T}} A W_k w = 0$ by (7). Thus $Z_k^{\mathrm{T}} r_k = Z_k^{\mathrm{T}} b - Z_k A x_k = Z_k^{\mathrm{T}} b$ and it follows that α is identical to Λ_k given by (4), and the multiparameter projection method of Section 1.2 is recovered.

Since condition (7) has to be satisfied $\forall \alpha$ and $\forall w$, we must have $W_k^{\mathrm{T}} A Z_k = 0$. One way to achieve this is to impose the conditions

$$Z_k^{\mathrm{T}} A Z_i = 0, \quad i \neq k. \tag{8}$$

In that case, the direction matrices Z_k are said to be *conjugate* (with respect to A if it is needed to be more precise). So, this method appears as a multiparameter generalization of the method of conjugate direction [17, p. 108] (which is exactly recovered if $\forall k$, $n_k = 1$). For that reason, it will be called the *multiparameter conjugate direction* method.

We have now to check if it is possible to find Z_k, conjugate to $Z_0, \ldots, Z_{k-1}$, such that $Z_k^{\mathrm{T}} A Z_k$ is nonsingular and satisfying $Z_k^{\mathrm{T}} r_k \neq 0$. Assume that $Z_0, \ldots, Z_{k-1}$ are conjugate. Since $x_k = W_k w$, then $\forall Z$, $Z^{\mathrm{T}} r_k = Z^{\mathrm{T}} b - Z^{\mathrm{T}} A W_k w$. If, $\forall Z$ conjugate to $Z_0, \ldots, Z_{k-1}$, we have $Z^{\mathrm{T}} r_k = 0$, then it follows that $Z^{\mathrm{T}} b = 0$ which means that $b \in AE_{k-1}$ or, in other words, $A^{-1} b = x \in E_{k-1}$. Thus, by the minimization property of x_k over E_{k-1}, we have $x_k = x$ and $r_k = 0$. Hence, if $r_k \neq 0$, we can find Z_k conjugate to $Z_0, \ldots, Z_{k-1}$ and satisfying $Z_k^{\mathrm{T}} r_k \neq 0$. Since $Z_k \neq 0$, it implies that $Z_k^{\mathrm{T}} A Z_k = (BZ_k)^{\mathrm{T}} BZ_k$ is positive definite (since Z_k is a full rank matrix) and, thus, nonsingular.

The condition $Z_k^{\mathrm{T}} A Z_i = 0, i \neq k$ means that, for any column z of Z_k and any column y of Z_i, we have $(y, Az) = 0$. Thus, when $i \neq k$, the columns of Z_i are linearly independent of those of Z_k and, since $Z_k^{\mathrm{T}} r_k \neq 0$, a finite termination property follows

Property 1. *Let us assume that there exists p such that $m_{p-1} = n$. Then, there exists $k \leqslant p$ such that $r_k = 0$ and $x_k = x$.*

Another property is the following.

Property 2.

$$Z_i^{\mathrm{T}} r_k = 0, \quad i = 0, \ldots, k-1.$$

Proof.

$$\begin{aligned} Z_0^{\mathrm{T}} r_1 &= Z_0^{\mathrm{T}} r_0 - Z_0^{\mathrm{T}} A Z_0 \Lambda_0 \\ &= Z_0^{\mathrm{T}} r_0 - Z_0^{\mathrm{T}} A Z_0 (Z_0^{\mathrm{T}} A Z_0)^{-1} Z_0^{\mathrm{T}} r_0 = 0. \end{aligned}$$

We assume that this property holds for the index k. We have

$$Z_i^{\mathrm{T}} r_{k+1} = Z_i^{\mathrm{T}} r_k - Z_i^{\mathrm{T}} A Z_k (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} r_k.$$

The first term on the right-hand side is zero for $i = 0, \ldots, k-1$ by the induction assumption and the second one by the conjugacy property. So $Z_i^{\mathrm{T}} r_{k+1} = 0$ for $i = 0, \ldots, k-1$. For $i = k$, we see that $Z_k^{\mathrm{T}} r_{k+1} = 0$ follows from the expression of Λ_k. $\square$

4. The multiparameter conjugate gradient algorithm

Let us now see how to construct matrices Z_k satisfying the conjugacy condition (8). As in the one-parameter case, such matrices are not uniquely determined and they can be obtained by various procedures. In this section, we will describe one of them.

From now on, we assume that $\forall k, n_k$ is constant and equal to v and we consider the matrices R_k and Z_k of dimension $n \times v$ constructed by the coupled two-term recurrence relationships

$$\begin{aligned} R_{k+1} &= R_k - A Z_k L_k, \quad L_k = (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}} R_k, \\ Z_{k+1} &= R_{k+1} + Z_k B_k. \end{aligned} \tag{9}$$

L_k and B_k are $v \times v$ matrices and $Z_0 = R_0$ is an arbitrary matrix. These two relations are similar to those used by O'Leary [23] in her block conjugate gradient algorithm.

We consider the hypothesis

$$\mathrm{H}: \quad Z_k - R_k \in \operatorname{span}(R_0, \ldots, R_{k-1}),$$

which means that $Z_k = R_k + \sum_{j=0}^{k-1} R_j c_j$ where the c_j's are $v \times v$ matrices.

It is easy to see that the matrices R_k and Z_k constructed by (9) satisfy the assumption H. Indeed, the assumption is true for $k = 0$. Assuming that it holds for k, we have

$$Z_{k+1} - R_{k+1} = Z_k B_k = [R_k + \mathrm{span}(R_0, \ldots, R_{k-1})]B_k \in \mathrm{span}(R_0, \ldots, R_k).$$

Let us now prove some properties of these two sets of matrices.

Property 3.

$$Z_i^{\mathrm{T}} R_k = 0, \quad i = 0, \ldots, k-1.$$

The proof is similar to that of Property 2.

Property 4. *Under the assumption H, we have,* $\forall i \neq k$, $R_i^{\mathrm{T}} R_k = 0$.

Proof. By Property 3, we have $\forall k \geqslant 1$, $Z_0^{\mathrm{T}} R_k = 0$. Since $R_0 = Z_0$, then, $\forall k \geqslant 1$, $R_0^{\mathrm{T}} R_k = 0$. Let us proceed by induction and assume that $R_i^{\mathrm{T}} R_k = 0$ for $i = 0, \ldots, k-2$. By H and Property 3, we have

$$Z_{i+1}^{\mathrm{T}} R_k = 0 = R_{i+1}^{\mathrm{T}} R_k + \sum_{j=0}^{i} c_j^{\mathrm{T}} R_j^{\mathrm{T}} R_k$$

for $i+1 = 0, \ldots, k-1$, i.e., for $i = 0, \ldots, k-2$. If $i = 0, \ldots, k-2$, the sum on the right-hand side is zero by the induction assumption and it follows that $R_{i+1}^{\mathrm{T}} R_k = 0$. Thus $R_i^{\mathrm{T}} R_k = 0$ for $i = 0, \ldots, k-1$. Since one of the indexes is always smaller than the other one, the result follows. □

It follows from Property 4 that

$$Z_k^{\mathrm{T}} R_k = R_k^{\mathrm{T}} R_k + \sum_{i=0}^{k-1} c_i^{\mathrm{T}} R_i^{\mathrm{T}} R_k = R_k^{\mathrm{T}} R_k.$$

So, we obtain another expression for L_k

$$L_k = (Z_k^{\mathrm{T}} A Z_k)^{-1} R_k^{\mathrm{T}} R_k. \tag{10}$$

We see that the matrix L_k is nonsingular if and only if R_k has rank v.

Property 5. *Under the assumption H and if*, $\forall k$, L_k *is nonsingular then*, $\forall k \geqslant 2$,

$$Z_i^{\mathrm{T}} A R_k = Z_i^{\mathrm{T}} A^2 Z_k = 0, \quad i = 0, \ldots, k-2.$$

Proof. From Property 4, we have $(R_{i+1} - R_i)^{\mathrm{T}} R_k = 0$ for $i = 0, \ldots, k-2$. Thus, for $i = 0, \ldots, k-2$, $L_i^{\mathrm{T}} Z_i^{\mathrm{T}} A R_k = 0$ and it follows that $Z_i^{\mathrm{T}} A R_k = 0$ since L_i is nonsingular. Therefore, for $i = 0, \ldots, k-2$, $Z_i^{\mathrm{T}} A(R_{k+1} - R_k) = Z_i^{\mathrm{T}} A A Z_k L_k = 0$ and, thus, $Z_i^{\mathrm{T}} A A Z_k = 0$ since L_k is nonsingular. □

Let us now look at the conjugacy property of the matrices Z_k constructed by (9).

From the expression of Z_1, we have $Z_0^T AZ_1 = Z_0^T AR_1 + Z_0^T AZ_0 B_0$ which is zero if $B_0 = -(Z_0^T AZ_0)^{-1} Z_0^T AR_1$. Then, we proceed by induction. We have

$$Z_i^T AZ_{k+1} = Z_i^T AR_{k+1} + Z_i^T AZ_k B_k.$$

The first term on the right-hand side is zero for $i = 0, \ldots, k-1$ by Property 5. The second term on the right-hand side is zero for $i = 0, \ldots, k-1$ by the induction hypothesis. So, the conjugacy property is satisfied for $i = 0, \ldots, k-1$.

For $i = k$, we have

$$Z_k^T AZ_{k+1} = Z_k^T AR_{k+1} + Z_k^T AZ_k^T B_k.$$

So, with the choice

$$B_k = -(Z_k^T AZ_k)^{-1} Z_k^T AR_{k+1} \tag{11}$$

the conjugacy property also holds for $i = k$. Since one of the indexes is always smaller than the other one, the conjugacy property is true for $i \neq k$.

Let us give another expression for B_k. We have, from Property 4,

$$Z_k^T AR_{k+1} = L_k^{-T} (R_k - R_{k+1})^T R_{k+1} = -L_k^{-T} R_{k+1}^T R_{k+1}.$$

But $Z_k B_k = Z_{k+1} - R_{k+1}$ and it follows

$$Z_k^T AZ_k B_k = Z_k^T A(Z_{k+1} - R_{k+1}) = -Z_k^T AR_{k+1} = L_k^{-T} R_{k+1}^T R_{k+1}.$$

Replacing L_k by its expression (10), we finally obtain

$$B_k = (R_k^T R_k)^{-1} R_{k+1}^T R_{k+1}. \tag{12}$$

Let us now construct R_0 by partitioning the vector r_0 into v subvectors as explained in Section 2. We have $r_0 = R_0 e$ with $e = (1, \ldots, 1)^T \in \mathbb{R}^v$. We also see that $\Lambda_k = L_k e$. Multiplying the vector e by the recurrence relationship for R_k, we get, $\forall k$, $r_k = R_k e$. Thus, expression (10) for L_k leads to a new expression for Λ_k by multiplication by the vector e

$$\Lambda_k = (Z_k^T AZ_k)^{-1} R_k^T r_k. \tag{13}$$

Therefore, we have obtained an algorithm which is completely similar to the conjugate gradient algorithm of Hestenes and Stiefel [18] and, for that reason, it will be called the *multiparameter conjugate gradient* algorithm.

Let us set $z_k = Z_k e$. Since $r_k = R_k e$, we immediately obtain from Properties 2–5.

Property 6.

$$\begin{aligned} r_i^T r_k &= 0, \qquad i \neq k, \\ R_k^T r_i &= 0, \qquad i \neq k, \\ z_i^T r_k &= 0, \qquad i = 0, \ldots, k-1, \end{aligned}$$

$$\begin{aligned}
Z_i^{\mathrm{T}} r_k &= R_k^{\mathrm{T}} z_i = 0, && i = 0, \ldots, k-1,\\
z_i^{\mathrm{T}} A r_k &= z_i^{\mathrm{T}} A^2 z_k = 0, && i = 0, \ldots, k-2,\\
Z_i^{\mathrm{T}} A r_k &= Z_i^{\mathrm{T}} A^2 z_k = 0, && i = 0, \ldots, k-2,\\
z_i^{\mathrm{T}} A R_k &= z_i^{\mathrm{T}} A^2 Z_k = 0, && i = 0, \ldots, k-2.
\end{aligned}$$

Remark 2. Although R_0 and Z_0 are obtained by partitioning the vector r_0 by the strategy described in Section 2 and the relations $r_k = R_k e$ and $z_k = Z_k e$ hold, the matrices R_k and Z_k are not identical to the matrices obtained by applying exactly the same partitioning to the vectors r_k and z_k.

Let us now give the complete algorithm. It must be noticed that the computation of L_k and B_k by formulae (10) and (12) needs the inversion of two $v \times v$ matrices while, using (10) and (11), requires only one. So, although it is not a drawback to use (10) and (12) since v is usually small, we will employ (10) and (11).

Initializations

Choose x_0
$r_0 = b - Ax_0$
partition r_0 and construct R_0
$Z_0 = R_0$

Iterations

For $k = 0, 1, \ldots$ until convergence
$\quad G_k = (Z_k^{\mathrm{T}} A Z_k)^{-1} Z_k^{\mathrm{T}}$
$\quad L_k = G_k R_k$
$\quad R_{k+1} = R_k - A Z_k L_k$
$\quad r_{k+1} = R_{k+1} e$
$\quad \Lambda_k = L_k e$
$\quad x_{k+1} = x_k + Z_k \Lambda_k$
$\quad B_k = -G_k A R_{k+1}$
$\quad Z_{k+1} = R_{k+1} + Z_k B_k$
end for

5. The multiparameter Lanczos method

Instead of expressing, in the multiparameter conjugate direction method, x_{k+1} in terms of x_k and r_{k+1} in terms of r_k, let us now express them in terms of x_0 and r_0 respectively.

From relations (2), we immediately see that we have

$$x_k = x_0 + W_k \widetilde{\Lambda}_k \tag{14}$$

with $W_k = [Z_0, \ldots, Z_{k-1}]$ and $\widetilde{\Lambda}_k = (\Lambda_0^{\mathrm{T}}, \ldots, \Lambda_{k-1}^{\mathrm{T}})^{\mathrm{T}}$. Moreover,

$$r_k = r_0 - AW_k\widetilde{\Lambda}_k$$

and Property 2 writes $W_k^{\mathrm{T}} r_k = 0$. Thus

$$\widetilde{\Lambda}_k = (W_k^{\mathrm{T}} AW_k)^{-1} W_k^{\mathrm{T}} r_0.$$

Thanks to the biconjugacy property of the matrices Z_k, we see that the matrix $W_k^{\mathrm{T}} AW_k$ is block diagonal and that its diagonal blocks are $Z_i^{\mathrm{T}} AZ_i$ for $i = 0, \ldots, k-1$. Thus, for recovering the expression (4) for Λ_k, it remains to prove that the components of the vector $W_k^{\mathrm{T}} r_0$ are equal to $Z_i^{\mathrm{T}} r_i$ for $i = 0, \ldots, k-1$. Setting $Q_i = I - P_i$ with $P_i = AZ_i(Z_i^{\mathrm{T}} AZ_i)^{-1} Z_i^{\mathrm{T}}$, we have $r_{i+1} = Q_i r_i$, that is $r_i = \widetilde{Q}_i r_0$ with $\widetilde{Q}_i = Q_{i-1} \cdots Q_0$ for $i \geqslant 1$ and $\widetilde{Q}_0 = I$. Thus, we need to prove that the components of $W_k^{\mathrm{T}} r_0$ are equal to $Z_i \widetilde{Q}_i r_0$, that is, in other words, $Z_i^{\mathrm{T}} \widetilde{Q}_i = Z_i^{\mathrm{T}}$. First, it must be noticed that, thanks to the conjugacy property,

$$Z_i^{\mathrm{T}} P_j = Z_i^{\mathrm{T}} AZ_j (Z_j^{\mathrm{T}} AZ_j)^{-1} Z_j^{\mathrm{T}} = \begin{cases} 0, & i \neq j, \\ Z_i^{\mathrm{T}}, & i = j. \end{cases}$$

Now, we have

$$\begin{aligned} Z_i^{\mathrm{T}} \widetilde{Q}_i &= Z_i^{\mathrm{T}} Q_{i-1} \cdots Q_0 \\ &= Z_i^{\mathrm{T}} (I - P_{i-1}) Q_{i-2} \cdots Q_0 \\ &= Z_i^{\mathrm{T}} Q_{i-2} \cdots Q_0 \end{aligned}$$

since $Z_i^{\mathrm{T}} P_{i-1} = 0$ as we just saw. With the same reasoning, we obtain $Z_i^{\mathrm{T}} \widetilde{Q}_i = Z_i^{\mathrm{T}} Q_0 = Z_i^{\mathrm{T}} (I - P_0) = Z_i^{\mathrm{T}}$ since $Z_i^{\mathrm{T}} P_0 = 0$. So, the components of the vector $\widetilde{\Lambda}_k$ are consistent with the expression (4).

Thus, we have proved that the sequence (x_k) is completely defined by the two conditions (that will be called the *Lanczos conditions*)

1. $x_k - x_0 \in E_{k-1} = \mathrm{span}(Z_0, \ldots, Z_{k-1})$,
2. $r_k \perp E_{k-1}$,

which shows that the multiparameter conjugate direction method enters into the general framework of projection methods as described, for example, in [25] or [16].

We will prove now that, in the case of the multiparameter conjugate gradient algorithm, the subspace E_{k-1} is a generalization of a Krylov subspace.

Let $K_k(A, R_0)$ be the subspace of $n \times v$ matrices of the form $\sum_{i=0}^{k-1} A^i R_0 \alpha_i$ with $\alpha_i \in \mathbb{R}^{v \times v}$. We will prove by induction that, $\forall k, R_k, Z_k \in K_{k+1}(A, R_0)$. Obviously, the property is true for $k = 0$ since $Z_0 = R_0$. Assuming that the property is true for the index $k-1$, it is easy to see that it also holds for k by using the recurrence relationships (9). Thus, it follows that any vector in E_{k-1} can be written as $\sum_{i=0}^{k-1} A^i R_0 \beta_i$ where $\beta_i \in \mathbb{R}^v$. Therefore, $E_{k-1} = K_k(A, R_0)\Lambda$ with

$\Lambda \in \mathbb{R}^{\nu}$ and, so, the Lanczos conditions show that the conjugate gradient algorithm is aimed at implementing a method quite similar to Lanczos'. Thus, this method (completely defined by the Lanczos conditions) will be called the *multiparameter Lanczos method*. Algorithms for its implementation in the nonsymmetric case and other recurrences will be discussed in Sections 7 and 8.

The second Lanczos condition can be written as $R_0^{\mathrm{T}} A^i r_k = 0$ for $i = 0, \ldots, k-1$. But $r_k = r_0 + \sum_{i=1}^{k} A^i R_0 \gamma_i$ with $\gamma_i \in \mathbb{R}^{\nu}$. This is a polynomial of degree k in A and depending on R_0. For any matrix $R \in \mathbb{R}^{n \times \nu}$ and any vectors $\gamma_i \in \mathbb{R}^{\nu}$, let $P_k(\xi R)$ be the vector polynomial $P_k(\xi R) = \sum_{i=0}^{k} \xi^i R \gamma_i$. Then $r_k = P_k(AR_0)$. Let $C_i = R_0^{\mathrm{T}} A^i R$, $i \geqslant 0$ and let C be the linear mapping on the space of vector polynomials defined by

$$C(\xi^i R) = C_i, \quad i = 0, 1, \ldots$$

With these notations, the second Lanczos condition becomes

$$C(\xi^i P_k(\xi R_0)) = 0 \quad \text{for } i = 0, \ldots, k-1.$$

This condition is similar to the condition obtained for the one-parameter Lanczos method and it shows that P_k can be considered as a formal vector orthogonal polynomial with respect to the linear mapping C.

Thus the matrix and the formal orthogonal polynomial approaches to the classical Lanczos method (see [9], for instance) have been generalized to the multiparameter case.

6. The nonsymmetric case

If the matrix A of the system is nonsymmetric, there are several ways for transforming it into a symmetric one and, then, applying the methods of Section 1. We will now examine such strategies (described in [3, pp. 97–108] and [4]) but only for the multiparameter projection method since the Richardson method becomes too much complicated with too many matrix–vector products and the Barzilai–Borwein method remains unchanged.

For each case, we will first consider the multiparameter projection method of Section 1, then the multiparameter conjugate direction method described in Section 3 and, finally, the multiparameter conjugate gradient method given in Section 4.

In all cases, although the matrix of the system changes, r_k will still be defined as $r_k = b - Ax_k$.

6.1. Normal residuals (NR)

We consider the system $A^{\mathrm{T}} Ax = A^{\mathrm{T}} b$. For the multiparameter projection method, we still have iterations of the form (2) but, now, with

$$\Lambda_k = [(AZ_k)^{\mathrm{T}} AZ_k]^{-1} (AZ_k)^{\mathrm{T}} r_k.$$

By solving the system $r_{k+1} = 0 = r_k - AZ_k\Lambda_k$ in the least squares sense, we see that this value of Λ_k also minimizes $\|r_{k+1}\|$ and that

$$\|r_{k+1}\|^2 = \|r_k\|^2 - ((AZ_k)^{\mathrm{T}} r_k, [(AZ_k)^{\mathrm{T}} AZ_k]^{-1} (AZ_k)^{\mathrm{T}} r_k).$$

Thus $\|r_{k+1}\| \leqslant \|r_k\|$ and the sequence $(\|r_k\|)$ decreases monotonously.

6.2. Normal equations (NE)

We set $x = A^{\mathrm{T}} x'$ and we consider the system $AA^{\mathrm{T}} x' = b$. The multiparameter projection method leads to iterates x'_k and, setting $x_k = A^{\mathrm{T}} x'_k$, we get the iterations

$$x_{k+1} = x_k + A^{\mathrm{T}} Z_k \Lambda_k, \quad r_{k+1} = r_k - AA^{\mathrm{T}} Z_k \Lambda_k$$

with

$$\Lambda_k = \left[(A^{\mathrm{T}} Z_k)^{\mathrm{T}} A^{\mathrm{T}} Z_k\right]^{-1} Z_k^{\mathrm{T}} r_k.$$

By solving the system $e_{k+1} = 0 = e_k - A^{\mathrm{T}} Z_k \Lambda_k$ in the least squares sense, we see that this value of Λ_k also minimizes $\|e_{k+1}\|$ and that

$$\|e_{k+1}\|^2 = \|e_k\|^2 - \left(Z_k^{\mathrm{T}} r_k, \left[(A^{\mathrm{T}} Z_k)^{\mathrm{T}} A^{\mathrm{T}} Z_k\right]^{-1} Z_k^{\mathrm{T}} r_k\right).$$

Thus $\|e_{k+1}\| \leqslant \|e_k\|$ and the sequence $(\|e_k\|)$ decreases monotonously.

We will now discuss two particular procedures for constructing the matrices Z_k.

Let $\|\cdot\|$ by any norm on $\mathbb{R}^n$. We consider the vector z_k defined by $(z_k, r_k) = \|r_k\|$. For example, if the norm is the Euclidean one, we have $z_k = r_k/\|r_k\|$. Choosing such a vector z_k corresponds, in the one-parameter case, to the *norm decomposition* method due to Gastinel [13,14]. Its multiparameter extension can be defined by partitioning the vector z_k and then constructing the matrix Z_k as explained in Section 2.

Similarly, the method of Kaczmarz [21] consists of taking $z_k = e_i$, $k + 1 = i$ (mod. n). Then, Z_k can be obtained by partitioning such a vector z_k.

These two procedures remain to be studied in the multiparameter case.

6.3. Expanded system (ES)

We consider the system $My = c$ with

$$M = \begin{pmatrix} 0 & A \\ A^{\mathrm{T}} & 0 \end{pmatrix}, \quad y = \begin{pmatrix} x' \\ x \end{pmatrix}, \quad c = \begin{pmatrix} b \\ b' \end{pmatrix}.$$

Unless the solution of the system $A^{\mathrm{T}} x' = b'$ is needed, the vector b' can be arbitrarily chosen.

The matrix of this system is symmetric but not positive definite. Since the symmetry is the only property needed for deriving the algorithm, the method can still be defined but, as stated above, J is no longer convex and it follows that the minimization and monotonicity properties are not satisfied for each iteration k. Moreover, the matrix to be inverted can be singular even if the matrix corresponding to Z_k in the new procedure has full rank. In this situation, known as *breakdown*, the algorithm has to be stopped or, if possible, modified.

We set

$$y_k = \begin{pmatrix} x'_k \\ x_k \end{pmatrix}, \quad r(y_k) = c - My_k = \begin{pmatrix} r_k = b - Ax_k \\ r'_k = b' - A^{\mathrm{T}}x'_k \end{pmatrix}, \quad \widetilde{r}_k = \begin{pmatrix} r'_k \\ r_k \end{pmatrix}.$$

An important point to notice is that $r(y_k) \neq \widetilde{r}_k$.

The projection method of Section 1 applied to the system $My = c$ leads to the iterations

$$y_{k+1} = y_k + \widetilde{Z}_k \widetilde{\Lambda}_k,$$
$$r(y_{k+1}) = r(y_k) - M\widetilde{Z}_k\widetilde{\Lambda}_k.$$

There are two possible choices for $\widetilde{Z}_k$ which lead to two different expressions for $\widetilde{\Lambda}_k$.

6.3.1. Extended system, first choice (ES1)

The first choice is

$$\widetilde{Z}_k = \begin{pmatrix} Z'_k \\ Z_k \end{pmatrix} \in \mathbb{R}^{2n \times n_k}$$

with $Z_k, Z'_k \in \mathbb{R}^{n \times n_k}$. In that case, $\widetilde{\Lambda}_k = \Lambda_k \in \mathbb{R}^{n_k}$ which leads to the iterations

$$x_{k+1} = x_k + Z_k\Lambda_k, \quad r_{k+1} = r_k - AZ_k\Lambda_k,$$
$$x'_{k+1} = x'_k + Z'_k\Lambda_k, \quad r'_{k+1} = r'_k - A^{\mathrm{T}}Z'_k\Lambda_k$$

with

$$\Lambda_k = (\widetilde{Z}_k^{\mathrm{T}} M \widetilde{Z}_k)^{-1}\widetilde{Z}_k^{\mathrm{T}} r(y_k)$$
$$= (Z_k'^{\mathrm{T}} A Z_k + Z_k^{\mathrm{T}} A^{\mathrm{T}} Z'_k)^{-1}(Z_k'^{\mathrm{T}} r_k + Z_k^{\mathrm{T}} r'_k).$$

These relations do not reduce to the usual ones in the one-parameter case since, in general, for arbitrary vectors z_k and z'_k, $(z'_k, r_k) \neq (z_k, r'_k)$.

6.3.2. Extended system, second choice (ES2)

The second choice is

$$\widetilde{Z}_k = \begin{pmatrix} Z'_k & 0 \\ 0 & Z_k \end{pmatrix} \in \mathbb{R}^{2n \times (n'_k + n_k)}$$

with $Z_k \in \mathbb{R}^{n \times n_k}$ and $Z'_k \in \mathbb{R}^{n \times n'_k}$, and we have

$$\widetilde{\Lambda}_k = \begin{pmatrix} \Lambda'_k \\ \Lambda_k \end{pmatrix} \in \mathbb{R}^{n'_k + n_k}.$$

For this case, we obtain the iterations

$$\begin{aligned} x_{k+1} &= x_k + Z_k \Lambda_k, \quad r_{k+1} = r_k - A Z_k \Lambda_k, \\ x'_{k+1} &= x'_k + Z'_k \Lambda'_k, \quad r'_{k+1} = r'_k - A^{\mathrm{T}} Z'_k \Lambda'_k \end{aligned}$$

with

$$\begin{aligned} \Lambda_k &= (Z'^{\mathrm{T}}_k A Z_k)^{-1} Z'^{\mathrm{T}}_k r_k, \\ \Lambda'_k &= (Z^{\mathrm{T}}_k A^{\mathrm{T}} Z'_k)^{-1} Z^{\mathrm{T}}_k r'_k. \end{aligned}$$

These relations do not reduce, in general, to the usual ones in the one-parameter case. This is due to the fact that, when $n_k = n'_k = 1$, $\widetilde{Z}_k$ is a $2n \times 2$ matrix while, in the one-parameter case, $\widetilde{Z}_k = (z'^{\mathrm{T}}_k, z^{\mathrm{T}}_k)^{\mathrm{T}} \in \mathbb{R}^{2n}$.

We see that the distinction between these two choices is that the stepsize vector is not the same in both recurrences for ES2, while it is the same for ES1. It follows that the expressions for $\widetilde{\Lambda}_k$ also change. Moreover, for ES2, the number of columns of the matrices Z_k and Z'_k can be different.

6.4. Positive definite expanded system (PDES)

We consider the system $My = c$ given by

$$\begin{pmatrix} A^{\mathrm{T}} A & 0 \\ 0 & A A^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix} = \begin{pmatrix} A^{\mathrm{T}} b \\ b \end{pmatrix}$$

with $x = A^{\mathrm{T}} x'$.

We set $w_k = A^{\mathrm{T}} x'_k$ and $s_k = b - A w_k$ and the projection method of Section 1 becomes, for the choice of $\widetilde{Z}_k$ considered in ES1,

$$\begin{aligned} x_{k+1} &= x_k + Z_k \Lambda_k, \quad r_{k+1} = r_k - A Z_k \Lambda_k, \\ w_{k+1} &= w_k + A^{\mathrm{T}} Z'_k \Lambda_k, \quad s_{k+1} = s_k - A A^{\mathrm{T}} Z'_k \Lambda_k \end{aligned}$$

with

$$\Lambda_k = [(A Z_k)^{\mathrm{T}} A Z_k + (A^{\mathrm{T}} Z'_k)^{\mathrm{T}} A^{\mathrm{T}} Z'_k]^{-1} ((A Z_k)^{\mathrm{T}} r_k + Z'^{\mathrm{T}}_k s_k).$$

For the choice of $\widetilde{Z}_k$ made in ES2, Λ'_k replaces Λ_k in the recurrence relationships for w_{k+1} and s_{k+1} and

$$\begin{aligned} \Lambda_k &= [(A Z_k)^{\mathrm{T}} A Z_k]^{-1} (A Z_k)^{\mathrm{T}} r_k, \\ \Lambda'_k &= [(A^{\mathrm{T}} Z'_k)^{\mathrm{T}} A^{\mathrm{T}} Z'_k]^{-1} Z'^{\mathrm{T}}_k s_k. \end{aligned}$$

We see that the products $AZ_k, A^{\mathrm{T}}Z'_k$ and $A(A^{\mathrm{T}}Z'_k)$ have to be computed. Since this procedure is more expensive than the previous one, we will not pursue its study.

7. Multiparameter biconjugate gradient algorithms

In this Section, we will follow the strategies described in Section 6 for applying the multiparameter conjugate gradient algorithm of Section 4 to a nonsymmetric linear system. Now on, we will assume that $\forall k, n_k = v$.

7.1. NR

Let us begin by the normal residuals. We have to replace A by $A^{\mathrm{T}}A$, r_k by $A^{\mathrm{T}}r_k$ and R_k by $A^{\mathrm{T}}R_k$. The iterates are still given by (2) but, now, with

$$\begin{aligned}\Lambda_k &= [(AZ_k)^{\mathrm{T}}AZ_k]^{-1}(AZ_k)^{\mathrm{T}}r_k\\ &= [(AZ_k)^{\mathrm{T}}AZ_k]^{-1}(A^{\mathrm{T}}R_k)^{\mathrm{T}}A^{\mathrm{T}}r_k.\end{aligned}$$

We also have

$$\begin{aligned}R_{k+1} &= R_k - AZ_kL_k,\\ Z_{k+1} &= A^{\mathrm{T}}R_{k+1} + Z_kB_k\end{aligned}$$

with $Z_0 = A^{\mathrm{T}}R_0$ and

$$\begin{aligned}L_k &= [(AZ_k)^{\mathrm{T}}AZ_k]^{-1}(AZ_k)^{\mathrm{T}}R_k\\ &= [(AZ_k)^{\mathrm{T}}AZ_k]^{-1}(A^{\mathrm{T}}R_k)^{\mathrm{T}}A^{\mathrm{T}}R_k,\\ B_k &= -[(AZ_k)^{\mathrm{T}}AZ_k]^{-1}(AZ_k)^{\mathrm{T}}AA^{\mathrm{T}}R_{k+1}\\ &= [(A^{\mathrm{T}}R_k)^{\mathrm{T}}A^{\mathrm{T}}R_k]^{-1}(A^{\mathrm{T}}R_{k+1})^{\mathrm{T}}A^{\mathrm{T}}R_{k+1}.\end{aligned}$$

This algorithm appears as a *multiparameter CGNR* algorithm. It was proposed by Hestenes and Stiefel [18] in the one-parameter case.

7.2. NE

Let us now apply the multiparameter conjugate gradient algorithm to the normal equations. We have to replace A by AA^{T}, we set $V_k = A^{\mathrm{T}}Z_k$ and we obtain

$$x_{k+1} = x_k + V_k\Lambda_k, \quad r_{k+1} = r_k - AV_k\Lambda_k,$$

with $\Lambda_k = (V_k^{\mathrm{T}}V_k)^{-1}V_k^{\mathrm{T}}r_k$. We also have

$$R_{k+1} = R_k - AV_kL_k, \quad L_k = (V_k^{\mathrm{T}}V_k)^{-1}V_k^{\mathrm{T}}R_k,$$
$$V_{k+1} = A^{\mathrm{T}}R_{k+1} + V_kB_k, \quad B_k = -(V_k^{\mathrm{T}}V_k)^{-1}V_k^{\mathrm{T}}R_{k+1}.$$

This algorithm appears as a *multiparameter CGNE* algorithm. It was proposed, in the one-parameter case, by Craig [11].

7.3. ES

Let us now consider the case of the extended system. The vectors x_k, x_k', r_k and r_k' are computed by formulae given in Section 6.3 for the two ES. In both cases, the hypothesis H becomes

$$\mathrm{H}: \quad \widetilde{Z}_k - \widetilde{R}_k \in \mathrm{span}(\widetilde{R}_0, \ldots, \widetilde{R}_{k-1})$$

with $\widetilde{Z}_0 = \widetilde{R}_0$ and the conjugacy becomes $\forall i \neq k$, $\widetilde{Z}_i^{\mathrm{T}}M\widetilde{Z}_k = 0$.

7.3.1. ES1

We first have the following property.

Property 7.

$$\forall k > 0, \widetilde{Z}_i^{\mathrm{T}}r(y_k) = 0, \quad i = 0, \ldots, k-1.$$

Proof. For $k = 1$, we have

$$\widetilde{Z}_0^{\mathrm{T}}r(y_1) = \widetilde{Z}_0^{\mathrm{T}}r(y_0) - \widetilde{Z}_0^{\mathrm{T}}M\widetilde{Z}_0\Lambda_0 = 0$$

by the expression for Λ_0. Assuming that the property is true for the index k, we have

$$\widetilde{Z}_i^{\mathrm{T}}r(y_{k+1}) = \widetilde{Z}_i^{\mathrm{T}}r(y_k) - \widetilde{Z}_i^{\mathrm{T}}M\widetilde{Z}_k\Lambda_k.$$

For $i = 0, \ldots, k-1$, the first term on the right-hand side is zero by the induction assumption and the second one by the conjugacy property. This expression is also zero for $i = k$ by definition of Λ_k. $\square$

Defining the matrices

$$\widetilde{I} = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}, \quad \widetilde{R}_k = \begin{pmatrix} R_k' \\ R_k \end{pmatrix}$$

we see that

$$\widetilde{I}\,\widetilde{R}_k = \begin{pmatrix} R_k \\ R_k' \end{pmatrix}$$

and relations (9) and (12) become

$$\begin{aligned} \widetilde{I}\,\widetilde{R}_{k+1} &= \widetilde{I}\,\widetilde{R}_k - M\widetilde{Z}_k\widetilde{L}_k, & \widetilde{L}_k &= (\widetilde{Z}_k^{\mathrm{T}} M\widetilde{Z}_k)^{-1}\widetilde{Z}_k^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_k, \\ \widetilde{Z}_{k+1} &= \widetilde{R}_{k+1} + \widetilde{Z}_k\widetilde{B}_k, & \widetilde{B}_k &= -(\widetilde{Z}_k^{\mathrm{T}} M\widetilde{Z}_k)^{-1}\widetilde{Z}_k^{\mathrm{T}} M\widetilde{R}_{k+1} \end{aligned}$$

with $Z_0 = R_0$ and $Z'_0 = R'_0$. In other terms, we have

$$\begin{aligned} R_{k+1} &= R_k - AZ_k\widetilde{L}_k, & R'_{k+1} &= R'_k - A^{\mathrm{T}} Z'_k\widetilde{L}_k, \\ Z_{k+1} &= R_{k+1} + Z_k\widetilde{B}_k, & Z'_{k+1} &= R'_{k+1} + Z'_k\widetilde{B}_k \end{aligned}$$

with

$$\begin{aligned} \widetilde{L}_k &= (Z_k'^{\mathrm{T}} AZ_k + Z_k^{\mathrm{T}} A^{\mathrm{T}} Z'_k)^{-1}(Z_k'^{\mathrm{T}} R_k + Z_k^{\mathrm{T}} R'_k), \\ \widetilde{B}_k &= (Z_k'^{\mathrm{T}} AZ_k + Z_k^{\mathrm{T}} A^{\mathrm{T}} Z'_k)^{-1}(Z_k'^{\mathrm{T}} AR_{k+1} + Z_k^{\mathrm{T}} A^{\mathrm{T}} R'_{k+1}). \end{aligned}$$

The hypothesis H is satisfied for $k = 0$. Assuming that it holds for k, we have $\widetilde{Z}_{k+1} - \widetilde{R}_{k+1} = \widetilde{Z}_k\widetilde{B}_k = [\widetilde{R}_k + \mathrm{span}(\widetilde{R}_0, \ldots, \widetilde{R}_{k-1})]\widetilde{B}_k \in \mathrm{span}(\widetilde{R}_0, \ldots, \widetilde{R}_k)$, which shows that it is satisfied for all k.

We have the following property.

Property 8.

$$\forall k > 0, \quad \widetilde{Z}_i^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_k = 0, \quad i = 0, \ldots, k-1.$$

Proof. We have

$$\widetilde{Z}_0^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_1 = \widetilde{Z}_0^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_0 - \widetilde{Z}_0^{\mathrm{T}} M\widetilde{Z}_0\widetilde{L}_0 = 0$$

by the expression of $\widetilde{L}_0$. Assuming that the property holds for the index k, we have

$$\widetilde{Z}_i^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_{k+1} = \widetilde{Z}_i^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_k - \widetilde{Z}_i^{\mathrm{T}} M\widetilde{Z}_k\widetilde{L}_k.$$

For $i = 0, \ldots, k-1$, the first term on the right-hand side is zero by induction and the second one by the conjugacy property. For $i = k$, we also obtain zero thanks to the expression of $\widetilde{L}_k$. $\square$

We also have the following results whose proofs, except for the introduction of the matrix $\widetilde{I}$, are quite similar to those of Section 4. So, they will be omitted.

Property 9.

$$\forall i \neq k, \quad \widetilde{R}_i^{\mathrm{T}}\widetilde{I}\,\widetilde{R}_k = 0.$$

Property 10.

$$\forall k > 1, \quad \widetilde{Z}_i^{\mathrm{T}} M\widetilde{R}_k = \widetilde{Z}_i^{\mathrm{T}} M\widetilde{I} M\widetilde{Z}_k = 0, \quad i = 0, \ldots, k-2.$$

It is easy to check that $\widetilde{Z}_i^{\mathrm{T}} M \widetilde{Z}_k = 0$ for $i \neq k$ and that we have the new expressions

$$\widetilde{L}_k = (\widetilde{Z}_k^{\mathrm{T}} M \widetilde{Z}_k)^{-1} \widetilde{R}_k^{\mathrm{T}} \widetilde{I} \widetilde{R}_k,$$
$$\widetilde{B}_k = (\widetilde{R}_k^{\mathrm{T}} \widetilde{I} \widetilde{R}_k)^{-1} \widetilde{R}_{k+1}^{\mathrm{T}} \widetilde{I} \widetilde{R}_{k+1}.$$

If $\widetilde{R}_0$ is constructed by partitioning the vector $\widetilde{r}_0$, then $\widetilde{r}_0 = \widetilde{R}_0 e$ and $r(y_0) = \widetilde{I} \widetilde{R}_0 e$. Thus $\Lambda_k = \widetilde{L}_k e$ and we have

$$\widetilde{I} \widetilde{R}_{k+1} e = \widetilde{I} \widetilde{R}_k e - M \widetilde{Z}_k \widetilde{L}_k e = r(y_k) - M \widetilde{Z}_k \Lambda_k = r(y_{k+1}).$$

Thus, $\forall k$, $\widetilde{r}_k = \widetilde{R}_k e$ and $r(y_k) = \widetilde{I} \widetilde{R}_k e$ and it follows from the preceding properties.

Property 11.

$$\forall k > 0, \quad \widetilde{Z}_i^{\mathrm{T}} r(y_k) = 0, \quad i = 0, \ldots, k-1,$$
$$\forall i \neq k, \quad \widetilde{r}_i^{\mathrm{T}} r(y_k) = 0,$$
$$\forall k > 1, \quad \widetilde{Z}_i^{\mathrm{T}} M \widetilde{r}_k = 0, \quad i = 0, \ldots, k-2.$$

Thus, we have obtained a *multiparameter biconjugate gradient* algorithm which generalizes the BiCG of Fletcher [12]. If $v = 1$, this algorithm is exactly recovered since

$$\widetilde{Z}_k^{\mathrm{T}} M \widetilde{Z}_k = Z_k'^{\mathrm{T}} A Z_k + Z_k^{\mathrm{T}} A^{\mathrm{T}} Z_k',$$
$$\widetilde{R}_k^{\mathrm{T}} r(\widetilde{x}_k) = R'^{\mathrm{T}} r_k + R_k^{\mathrm{T}} r_k',$$
$$\widetilde{R}_k^{\mathrm{T}} \widetilde{I} \widetilde{R}_k = R_k'^{\mathrm{T}} R_k + R_k^{\mathrm{T}} R_k'.$$

7.3.2. ES2

We set

$$\widetilde{R}_k = \begin{pmatrix} R_k' & 0 \\ 0 & R_k \end{pmatrix}$$

and relations (9) and (12) become

$$\begin{aligned} R_{k+1} &= R_k - A Z_k L_k, & L_k &= (Z_k'^{\mathrm{T}} A Z_k)^{-1} Z_k'^{\mathrm{T}} R_k, \\ R_{k+1}' &= R_k' - A^{\mathrm{T}} Z_k' L_k', & L_k' &= (Z_k^{\mathrm{T}} A^{\mathrm{T}} Z_k')^{-1} Z_k^{\mathrm{T}} R_k', \\ Z_{k+1} &= R_{k+1} + Z_k B_k, & B_k &= (Z_k'^{\mathrm{T}} A Z_k)^{-1} Z_k'^{\mathrm{T}} A R_{k+1}, \\ Z_{k+1}' &= R_{k+1}' + Z_k' B_k', & B_k' &= Z_k^{\mathrm{T}} A^{\mathrm{T}} Z_k')^{-1} Z_k^{\mathrm{T}} A^{\mathrm{T}} R_{k+1}', \end{aligned}$$

with $Z_0 = R_0$ and $Z_0' = R_0'$.

We have the following results.

Property 12.

$$
\begin{aligned}
R_i^{\mathrm{T}} R'_k &= 0, \quad i \neq k, \\
Z_i^{\mathrm{T}} R'_k &= 0, \quad i = 0, \ldots, k-1, \\
Z_i'^{\mathrm{T}} R_k &= 0, \quad i = 0, \ldots, k-1, \\
Z_i^{\mathrm{T}} A^{\mathrm{T}} R'_k &= 0, \quad i = 0, \ldots, k-2, \\
Z_i'^{\mathrm{T}} A R_k &= 0, \quad i = 0, \ldots, k-2, \\
Z_i^{\mathrm{T}} A^{\mathrm{T}} A^{\mathrm{T}} Z'_k &= 0, \quad i = 0, \ldots, k-2, \\
Z_i'^{\mathrm{T}} A A Z_k &= 0, \quad i = 0, \ldots, k-2.
\end{aligned}
$$

The proofs are quite similar to those of Section 4 and they will be omitted (see also [3, pp. 113–119]). They use the fact, easy to check from the recurrence relationships, that the assumption H is satisfied by the pairs of matrices (Z_k, R_k) and (Z'_k, R'_k).

Since R_0 and R'_0 are obtained by partitioning the vectors r_0 and r'_0 respectively, then $r_k = R_k e$ and $r'_k = R'_k e$ and it follows.

Property 13.

$$
\begin{aligned}
r_i^{\mathrm{T}} r'_k &= 0, \quad i \neq k, \\
Z_i^{\mathrm{T}} r'_k &= 0, \quad i = 0, \ldots, k-1, \\
Z_i'^{\mathrm{T}} r_k &= 0, \quad i = 0, \ldots, k-1, \\
Z_i^{\mathrm{T}} A^{\mathrm{T}} r'_k &= 0, \quad i = 0, \ldots, k-2, \\
Z_i'^{\mathrm{T}} A r_k &= 0, \quad i = 0, \ldots, k-2.
\end{aligned}
$$

Obviously, we can set $z_k = Z_k e$ and $z'_k = Z'_k e$, and derive, from Property 12, results similar to those of Property 6.

Let us now give other expressions for the matrices L_k, L'_k, B_k and B'_k. Again, the proofs are similar to those of Section 4 and they will be omitted.

First, we have

$$Z_k'^{\mathrm{T}} R_k = R_k'^{\mathrm{T}} R_k, \quad Z_k^{\mathrm{T}} R'_k = R_k^{\mathrm{T}} R'_k$$

and, thus,

$$L_k = (Z_k'^{\mathrm{T}} A Z_k)^{-1} R_k'^{\mathrm{T}} R_k, \quad L'_k = (Z_k^{\mathrm{T}} A^{\mathrm{T}} Z'_k)^{-1} R_k^{\mathrm{T}} R'_k.$$

Since $r_k = R_k e$ and $r'_k = R'_k e$, new expressions for Λ_k and Λ'_k follow by replacing R_k by r_k in L_k and R'_k by r'_k in L'_k.

Then, we can prove that

$$B_k = (R_k'^{\mathrm{T}} R_k)^{-1} R_{k+1}'^{\mathrm{T}} R_{k+1}, \quad B'_k = (R_k^{\mathrm{T}} R'_k)^{-1} R_{k+1}^{\mathrm{T}} R'_{k+1}.$$

Since the iterates x'_k are not computed (unless the solution of the system $A^{\mathrm{T}}x' = b'$ is needed), r'_0 is arbitrary and, so, R'_0 can be any $n \times v$ matrix Y.

Thus, we have obtained a *multiparameter biconjugate gradient* algorithm which generalizes the BiCG of Fletcher [12] (which is exactly recovered if $v = 1$).

7.4. PDES

Similarly, a *multiparameter biconjugate residuals* algorithm can be obtained from the positive definite extended system and similar properties can be proved. However, since, as stated above, this algorithm needs too many matrix products by A and A^{T}, we will not give it here.

8. Other multiparameter Lanczos type algorithms

It is well know that, in the one-parameter case, the recurrence relationships of the biconjugate gradient algorithm (also known as Lanczos/Orthomin) can be replaced by other recurrences thus leading to algorithms named Lanczos/Orthores, Lanczos/Orthodir, …, see [9]. In this Section, we will derive similar algorithms from the multiparameter biconjugate gradient algorithm in the case of the system ES2. The case of the system ES1 could be treated similarly. We will follow the same procedure as in the one-parameter case and the proofs will be quite similar to those of this case [4; 3, pp. 119–121]. Obviously, the following algorithms also apply to the symmetric positive definite case after deleting the recurrences for r'_k, R'_k and Z'_k.

We have $Z_k = R_k + Z_{k-1}B_{k-1}$ and $R_k = R_{k+1} + AZ_kL_k$. Thus $Z_k = R_{k+1} + AZ_kL_k + Z_{k-1}B_{k-1}$. But $R_{k+1} = Z_{k+1} - Z_kB_k$ and we finally obtain

$$Z_{k+1} = Z_k(I + B_k) - AZ_kL_k - Z_{k-1}B_{k-1}. \tag{15}$$

Obviously, a similar relation holds for Z'_{k+1}. Now, it must be noticed that if Z_k and Z'_k are replaced by Z_kc_k and $Z'_kc'_k$ respectively, where c_k and c'_k are nonsingular $v \times v$ matrices, then the vectors and the matrices computed by the multiparameter biconjugate gradient algorithm remain unchanged. Thus (15) can be replaced by

$$Z_{k+1} = AZ_k + Z_kU_k - Z_{k-1}V_k, \quad Z'_{k+1} = A^{\mathrm{T}}Z'_k + Z'_kU'_k - Z'_{k-1}V'_k.$$

Let us now compute the matrices U_k and V_k. We have

$$Z_i'^{\mathrm{T}}AZ_{k+1} = Z_i'^{\mathrm{T}}AAZ_k + Z_i'^{\mathrm{T}}AZ_kU_k - Z_i'^{\mathrm{T}}AZ_{k-1}V_k.$$

The first term on the right-hand side is zero for $i = 0, \ldots, k-2$ by Property 12. By the biconjugacy property, the second term is zero for $i = 0, \ldots, k-1$ and the third one for $i = 0, \dots, k-2$. So, the left hand side is zero for $i = 0, \dots, k-2$. For $i = k-1$ and $i = k$, we obtain

$$V_k = (Z'^{\mathrm{T}}_{k-1} A Z_{k-1})^{-1} Z'^{\mathrm{T}}_{k-1} A A Z_k, \quad U_k = -(Z'^{\mathrm{T}}_k A Z_k)^{-1} Z'^{\mathrm{T}}_k A A Z_k.$$

Similarly, we have

$$V'_k = (Z^{\mathrm{T}}_{k-1} A^{\mathrm{T}} Z'_{k-1})^{-1} Z^{\mathrm{T}}_{k-1} A^{\mathrm{T}} A^{\mathrm{T}} Z'_k, \quad U'_k = -(Z^{\mathrm{T}}_k A^{\mathrm{T}} Z'_k)^{-1} Z^{\mathrm{T}}_k A^{\mathrm{T}} A^{\mathrm{T}} Z'_k.$$

This algorithm appears as a *multiparameter Lanczos/Orthodir* algorithm.

Since the matrices Z_k and Z'_k are intermediate results, they can be eliminated from the algorithm. We have $R_{k+1} = R_k - AZ_k L_k$ and $AZ_k = AR_k + AZ_{k-1} B_{k-1}$. Thus $R_{k+1} = R_k - AR_k L_k - AZ_{k-1} B_{k-1} L_k$. But $AZ_{k-1} = (R_{k-1} - R_k) L_{k-1}^{-1}$ and we finally obtain

$$R_{k+1} = R_k (I + L_{k-1}^{-1} B_{k-1} L_k) - AR_k L_k - R_{k-1} L_{k-1}^{-1} B_{k-1} L_k. \tag{16}$$

A similar relation holds for R'_{k+1}.

We have to find new expressions for L_k and L'_k not involving the matrices Z_k and Z'_k. The relation (16) has the form

$$R_{k+1} = [AR_k + R_k S_k + R_{k-1} T_k] F_k$$

for some matrix F_k.

From Property 12, $R'^{\mathrm{T}}_{k-1} R_{k+1} = 0$ and $R'^{\mathrm{T}}_k R_{k+1} = 0$ and it follows

$$T_k = -(R'^{\mathrm{T}}_{k-1} R_{k-1})^{-1} R'^{\mathrm{T}}_{k-1} A R_k, \quad S_k = -(R'^{\mathrm{T}}_k R_k)^{-1} R'^{\mathrm{T}}_k A R_k.$$

From (16), we see that we must have $(S_k + T_k) F_k = I$ and thus $F_k = (S_k + T_k)^{-1}$. Similarly, it holds

$$R'_{k+1} = [A^{\mathrm{T}} R'_k + R'_k S'_k + R'_{k-1} T'_k] F'_k$$

with

$$T'_k = -(R^{\mathrm{T}}_{k-1} R'_{k-1})^{-1} R^{\mathrm{T}}_{k-1} A^{\mathrm{T}} R'_k, \quad S'_k = -(R^{\mathrm{T}}_k R'_k)^{-1} R^{\mathrm{T}}_k A^{\mathrm{T}} R'_k$$

and $F'_k = (S'_k + T'_k)^{-1}$.

This algorithm is a *multiparameter Lanczos/Orthores* algorithm. However, it must be noticed that it does not seem possible to avoid using the matrices Z_k in (2).

Acknowledgements

I would like to thank Florence Bantegnies for several corrections and pertinent remarks.

References

[1] J. Barzilai, J.M. Borwein, Two-point step size gradient methods, IMA J. Numer. Anal. 8 (1988) 141–148.
[2] C. Brezinski, Variations on Richardson's method and acceleration, in: Numerical Analysis. A Numerical Analysis Conference in Honor of Jean Meinguet, Bull. Soc. Math. Belg. (1996) 33–44.
[3] C. Brezinski, Projection Methods for Systems of Equations, North–Holland, Amsterdam, 1997.
[4] C. Brezinski, Projection methods for linear systems, J. Comput. Appl. Math. 77 (1997) 35–51.
[5] C. Brezinski, Vector sequence transformations: methodology and applications to linear systems, J. Comput. Appl. Math. 98 (1998) 149–175.
[6] C. Brezinski, J.P. Chehab, Nonlinear hybrid procedures and fixed point iterations, Numer. Funct. Anal. Optimization 19 (1998) 465–487.
[7] C. Brezinski, J.P. Chehab, Multiparameter iterative schemes for the solution of systems of linear and nonlinear equations, SIAM J. Sci. Comput., in press.
[8] C. Brezinski, M. Redivo Zaglia, Vector and matrix sequence transformations based on biorthogonality, Appl. Numer. Math. 21 (1996) 353–373.
[9] C. Brezinski, H. Sadok, Lanczos type algorithms for solving systems of linear equations, Appl. Numer. Math. 11 (1993) 443–473.
[10] S.L. Campbell, C.D. Meyer, Jr., Generalized Inverses of Linear Transformations, Pitman, London, 1979.
[11] E.J. Craig, The N-step iteration procedures, J. Math. Phys. 34 (1955) 64–73.
[12] R. Fletcher, Conjugate gradient methods for indefinite systems, in: G.A. Watson (Ed.), Numerical Analysis, Dundee 1975, LNM Vol. 506, Springer, Berlin, 1976, pp. 73–89.
[13] N. Gastinel, Procédé itératif pour la résolution numérique d'un système d'équations linéaires, C.R. Acad. Sci. Paris 246 (1958) 2571–2574.
[14] N. Gastinel, Sur–décomposition de normes générales et procédés itératifs, Numer. Math. 5 (1963) 142–151.
[15] G.H. Golub, C.F. Van Loan, Matrix Computations, 2nd edn, The Johns Hopkins University Press, Baltimore, 1989.
[16] A. Greenbaum, Iterative Methods for Solving Linear Systems, SIAM, Philadelphia, 1997.
[17] M.R. Hestenes, Conjugate Direction Methods in Optimization, Springer, Berlin, 1980.
[18] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Stand. 49 (1952) 409–436.
[19] A.S. Householder, The Theory of Matrices in Numerical Analysis, Blaisdell Publ., New York, 1964.
[20] A.S. Householder, F.L. Bauer, On certain iterative methods for solving linear systems, Numer. Math. 2 (1960) 55–59.
[21] S. Kaczmarz, Angenäherte Auflösung von Systemen linearer Gleichungen, Bull. Acad. Polon. Sci. A 35 (1937) 355–357.
[22] B. Molina, M. Raydan, Preconditioned Barzilai–Borwein method for the numerical solution of partial differential equations, Numerical Algorithms 13 (1996) 45–60.
[23] D.P. O'Leary, The block conjugate gradient algorithm and related methods, Linear Algebra Appl. 29 (1980) 293–322.
[24] M. Raydan, On the Barzilai and Borwein choice of steplength for the gradient method, IMA J. Numer. Anal. 13 (1993) 321–326.
[25] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS Publ., Boston, 1996.